IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Gibson et al.
Application No. 10/714,541
Filed: November 14, 2003
Confirmation No. 2536
For: A UNIVERSAL PARSING AGENT SYSTEM AND METHOD
Examiner: Robinson, Greta Lee
Art Unit: 2169
Attorney Reference No. 13938-E

CERTIFICATE OF ELECTRONIC FILING

I hereby certify that this paper and the documents referred to as being attached or enclosed herewith are being submitted to the United States Patent and Trademark Office using the electronic filing system in accordance with the Office electronic filing system requirements as set forth at 37 CFR § 1.6 on the date set forth below.

/ Douglas E. McKinley, Jr. /
Douglas E. McKinley, Jr.
Attorney or Agent
for Applicant(s)

Date Filed September 26, 2010

MAIL STOP APPEAL BRIEF - PATENTS
COMMISSIONER FOR PATENTS
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450

APPEAL BRIEF

This is an appeal brief filed under 37 C.F.R. §41.37. Applicants believe the Notice of Appeal was received by the U.S. Patent and Trademark Office (USPTO) on July 26, 2010, making the Appeal Brief due on September 27, 2009.

Real Party in Interest

The real party in interest is the Battelle Memorial Institute, the assignee of the present application.

Related Appeals and Interferences

There are no related Appeals or Interferences.

Status of Claims

Claims 1-22, 29-31 are pending. Claims 23-28 and 32 have been cancelled. Claims 1-22, 29-31 have been rejected, and are appealed.

Status of Amendments

No amendments have been filed subsequent to the final rejection.

Summary of Claimed Subject Matter

Generally speaking, the invention is a computer system for converting data from one or more information sources into a common format. The invention differs from the prior art, and provides a significant advancement over the prior art, because the invention allows the data to be put into the system from multiple digital formats, including unstructured data formats, and then to convert any or all of that data into a common format. The invention is able to do this, in part, because the invention allows for interaction with the system through a graphical user interface (GUI) to select the appropriate pattern for any given data set to achieve the desired end. Either directly, or by virtue of dependency, all pending claims contain this limitation which is generally recited as "receiving at least one pattern descriptor selected from a graphical user interface."

The examiner has relied exclusively on the teachings of Webber, US Patent No. 5,909,570 and Lennon, US Patent No. 7,287,018 as the basis for all rejections. At a minimum, the examiner's rejection is flawed because neither Webber or Lennon teach the limitation of "receiving at least one pattern descriptor selected from a graphical user interface." Instead, at best, both Webber and Lennon teach tangentially similar steps that occur in an automated fashion. As such, the combination of Webber and Lennon do not teach at least this limitation, as the limitation does not exist in either reference.

Independent Claim 1 relates to a method for extracting and converting data in a computer system from one or more information sources into a common format. The method includes the steps of receiving the information sources in a computer system; receiving at least one pattern descriptor selected from a graphical user interface; receiving one or more templates, each of the templates having at least one of the pattern descriptors; applying one or more templates to the information sources; generating the data in a common format by parsing the information sources with a universal parsing agent that utilizes the one or more templates; and storing the data in the common format. A description of these limitations is found at page 5, lines 1-9. A graphic description is shown in figures 1, 3 and 4. A graphic description of an embodiment using a server/client system is shown in figure 2. A detailed discussion of the universal parsing agent is found at page 14, lines 13-21, and in the flowchart of Figure 4, which is described at page 15, lines 14-21. The interaction between the universal parsing agent and the templates is described at page 15, lines 1-13.

Claim 2 is dependant from Claim 1, and adds the additional limitation that after storing the data in the common format, the method further comprises communicating the data to an application configured to process the common format. Support for this limitation is shown at page 15, lines 10-11.

Claim 3 is dependant from Claim 2, and adds the additional limitation that the application is a database application. Support for this limitation is shown at page 15, lines 10-11.

Claim 4 is dependant from Claim 1, and adds the additional limitation that common format for the structured data is a Extensible Markup Language (XML) format. Support for this limitation is shown at page 16, line 11 through page 17 line 7.

Claim 5 is dependant from Claim 1, and adds the additional limitation that before receiving the one or more templates, the method further comprises the steps of:

generating said one or more templates by selecting a file from said information sources, and

having a user select one or more pattern descriptors to describe said file.

Support for this limitation is shown at page 16, line 11 through page 17 line 7.

Claim 6 is dependant from Claim 5, and adds the additional limitation of permitting the user to define the one or more pattern descriptors. Support for this limitation is shown at page 16, lines 1-10.

Claim 7 is dependant from Claim 1, and adds the additional limitation that before receiving the one or more templates, the method further comprises permitting the user to select one or more templates from a template library. Support for this limitation is shown at page 16, lines 8-10.

Claim 8 is dependant from Claim 1, and adds the additional limitation that storing of the data in the common format is selected from a group of storage bins consisting of an input bin, a wait bin, an incomplete bin, and a complete bin. Support for this limitation is shown at page 17, lines 18 -21, and page 19, lines 3-18.

Independent Claim 9 relates to a system for extracting and converting data from one or more information sources into a common format. The system includes a memory configured to receive the information sources, the memory configured to store one or more templates wherein each of the templates has at least one pattern descriptor selected from a graphical user interface. The system further includes an input device configured to receive at least one pattern descriptor from a user interacting with a graphical user interface. The system further includes a processor programmed to apply the one or more templates to said information sources; generate the data in the common format by parsing the information sources with a universal parsing agent that utilizes the one or more templates; and communicate the data in the common format. A description of these limitations is found at page 5, lines 10-18. A graphic description is shown in figures 1, 3 and 4. A graphic description of an embodiment using a server/client system is shown in figure 2. A detailed discussion of the universal parsing agent is found at page 14, lines 13-21, and in the flowchart of Figure 4, which is described at page 15, lines 14-21. The interaction between the universal parsing agent and the templates is described at page 15, lines 1-13.

Claim 10 is dependant from Claim 9, and adds the additional limitation that after storing the data in the common format, the method further comprises communicating the data to an application configured to process the common format. Support for this limitation is shown at page 15, lines 10-11.

Claim 12 is dependant from Claim 9, and adds the additional limitation that common format for the structured data is a Extensible Markup Language (XML) format. Support for this limitation is shown at page 16, line 11 through page 17 line 7.

Claim 13 is dependant from Claim 9, and adds the additional limitation that the memory stores a template library from which a user can select one or more templates. Support for this limitation is shown at page 16, line 11 through page 17 line 7.

Claim 14 is dependant from Claim 9, and adds the additional limitation that the memory stores the data in a common format in a storage bin selected from a group of storage bins consisting of an input bin, a wait bin, and incomplete bin, and a complete bin. Support for this limitation is shown at page 17, lines 18 -21, and page 19, lines 3-18.

Independent claim 15 is directed to a computer readable medium encoded with a computer program having computer-executable instructions for performing a method for extracting and converting data from one or more information sources into a common format. The computer instructions configure a computer to receive the information sources in a computer system; receive at least one pattern descriptor selected from a graphical user interface; receive one or more templates, each of the templates having at least one of the pattern descriptors; applying one or more templates to the information sources; generating the data in a common format by parsing the information sources with a universal parsing agent that utilizes the one or more templates; and storing the data in the common format. A description of these limitations is found at page 5, lines 1-9, and a description of the computer readable medium that may be encoded with these instructions is found at page 13. A graphic description is shown in figures 1, 3 and 4. A graphic description of an embodiment using a server/client system is shown in figure 2. A detailed discussion of the universal parsing agent is found at page 14, lines 13-21, and in the flowchart of Figure 4, which is described at page 15, lines 14-21. The interaction between the universal parsing agent and the templates is described at page 15, lines 1-13.

Claim 16 is dependant from Claim 15, and adds the additional limitation that after storing the data in the common format, the method further comprises communicating the data to an application configured to process the common format. Support for this limitation is shown at page 15, lines 10-11.

Claim 17 is dependant from Claim 16, and adds the additional limitation that the application is a database application. Support for this limitation is shown at page 15, lines 10-11.

Claim 18 is dependant from Claim 15, and adds the additional limitation that common format for the structured data is a Extensible Markup Language (XML) format. Support for this limitation is shown at page 16, line 11 through page 17 line 7.

Claim 19 is dependant from Claim 15, and adds the additional limitation that before receiving the one or more templates, the method further comprises the steps of:

generating said one or more templates by selecting a file from said information sources, and

having a user select one or more pattern descriptors to describe said file.

Support for this limitation is shown at page 16, line 11 through page 17 line 7.

Claim 20 is dependant from Claim 16, and adds the additional limitation of permitting the user to define the one or more pattern descriptors. Support for this limitation is shown at page 16, lines 1-10.

Claim 21 is dependant from Claim 15, and adds the additional limitation that before receiving the one or more templates, the method further comprises permitting the user to select one or more templates from a template library. Support for this limitation is shown at page 16, lines 8-10.

Claim 22 is dependant from Claim 15, and adds the additional limitation that the storing of the data in a common format is in a storage bin selected from a group of storage bins consisting of an input bin, a wait bin, and incomplete bin, and a complete bin. Support for this limitation is shown at page 17, lines 18 –21, and page 19, lines 3-18.

Claim 29 is dependant from Claim 1, and adds the additional limitation that the information sources are selected from the group of structured information sources, semi-structured information sources, unstructured information sources and combinations thereof. Support for this limitation is shown at the paragraph bridging pages 7 and 8, the paragraph on page 8, lines 2-8, the paragraph bridging pages 13 and 14, and the paragraph on page 14, lines 5 - 14.

Claim 30 is dependant from Claim 9, and adds the additional limitation that the information sources are selected from the group of structured information sources, semi-structured information sources, unstructured information sources and combinations thereof. Support for this limitation is shown at the paragraph bridging pages 7 and 8, the paragraph on page 8, lines 2-8, the paragraph bridging pages 13 and 14, and the paragraph on page 14, lines 5 - 14.

Claim 31 is dependant from Claim 15, and adds the additional limitation that the information sources are selected from the group of structured information sources, semi-structured information sources, unstructured information sources and combinations thereof. Support for this limitation is shown at the paragraph bridging pages 7 and 8, the paragraph on page 8, lines 2-8, the paragraph bridging pages 13 and 14, and the paragraph on page 14, lines 5 - 14.

Grounds of Rejection to be Reviewed on Appeal

(1) Rejection of claims 1-22 and 29-31 under 35 U.S.C. § 103(a) as allegedly being unpatentable over Webber, US Patent No. 5,909,570 in view of Lennon, US Patent No. 7,287,018.

Argument

(1) Claims 1-22 and 29-31 are patentable under 35 U.S.C. § 103(a) over Webber, US Patent No. 5,909,570 in view of Lennon, US Patent No. 7,287,018.

The examiner has rejected claims 1-22 and 29-31 under 35 U.S.C. § 103(a) and has asserted that the claims are unpatentable under 35 U.S.C. § 103(a) over Webber, US Patent No. 5,909,570 in view of Lennon, US Patent No. 7,287,018. The examiner relies on Webber for all claimed features in claims 1-7, 9-14, 16-21 and 29-31 except that the examiner concedes that Webber does not teach that "the pattern descriptor is selected." The examiner asserts that Lennon provides this teaching.

With respect to claims 8, 14, and 22 the examiner additionally concedes that Webber does not teach "storage bins consisting of an input bin, a wait bin, an incomplete bin and a complete bin." The examiner then asserts that Lennon provides this teaching.

An invention that would not have been obvious to a person of ordinary skill at the time of the invention is patentable. 35 U.S.C. 103(a). As reiterated by the Supreme Court in *KSR International Co. v. Teleflex Inc.* (KSR), 550 U.S. ___, 82 USPQ2d 1385 (2007), the framework for the objective analysis for determining obviousness under 35 U.S.C. 103 is stated in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966). Obviousness is a question of law based on underlying factual inquiries. The factual inquiries enunciated by the Court are as follows:

(A) Determining the scope and content of the prior art,

(B) Ascertaining the differences between the claimed invention and the prior art; and

(C) Resolving the level of ordinary skill in the pertinent art.

The Scope and Content of the Prior Art

The fundamental problem with the examiner's rejection of the claims is that the examiner repeatedly misapprehends, or mischaracterizes, the teaching of Webber and Lennon. As such, the examiner's analysis of the facts at issue in determining obviousness is flawed.

While Webber is solving the same type of problem as the present invention, the specific problem Webber solves is much more narrow than the problem solved by the present invention, and the approach used by Webber is fundamentally different than the present invention. The main difference between the problem addressed by Webber and the problem addressed by the present invention is that the Webber is concerned with translating data between a first, pre-defined format which is recognized by the Webber system, into a second format, which is also pre-defined and recognized by the Webber system, yet which is dissimilar from the first format. Webber makes this plain in the background section of the Webber patent at column 1, lines 35-55 where Webber describes the problem Webber's invention is designed to solve:

> In order to ensure both the sender and recipient format data in a common mutually agreed way EDI takes place pursuant to a variety of established EDI standards. Examples of such standards (and their maintenance organizations) are the ANSI X12 standard (developed by the American National Standards Institute's Accredited Standards Committee's X12 group), the UN-EDI-FACT standard (Electronic Data Interchange for Administration, Commerce, and Transport, a United Nations standard based on ANSI X12 and the Trade Data Interchange standards used in Europe), the Uniform Communications Standards ("UCS"), and TDCC (developed by the Transportation Data Coordinating Committee).
>
> Regardless of the particular standard, all are designed around an electronic representation of a paper business document. A unique identifier code is assigned for each type of business document. For example, in the ANSI X12 standard an invoice is referred to as document number X12.2, with a transaction set identification code of 810.

In contrast to the known, standard, data types that Webber is manipulating, the present invention is directed towards translating data sets that may include data sets that are NOT pre-defined or recognized by the system into a common format. As a result of this distinction between these two problems, the present invention is compelled to take a decidedly different approach than that taken by Webber. Stated simply, the approach taken by Webber would not work for translating data sets whose structures were not pre-defined and which were not already recognized by the Webber system.

At column 10, lines 3-7 Webber recites that "…the template mapping system 10 is capable of performing a robust, flexible translation of a fixed-structure inbound dataset in the following manner…" It is thus explicit that Webber is manipulating data in a fixed-structure, and that the method demonstrated by Webber does not contemplate the unstructured data sources that the present invention is configured to manage.

As one example of how the examiner has misconstrued the facts in Webber, the template mapping system of Webber is not configured to use a pattern descriptor, because, for the reasons described immediately above, the Webber method is configured in advance to recognize the patterns in the data structures of the data used as input. Nevertheless, the examiner argues that Webber provides the teaching of the step of "receiving one or more templates, each of said templates having said at least one pattern descriptor." The examiner references col. 6 line 65 through col. 7 line 9 as providing this teaching, and recites the language "the template mapping system 10 of the present invention, and is thereby restructured and reformatted to be compatible with receiving."

However, a plain reading of this quoted language does not support the examiner's assertion. In this section of the Webber patent, Webber describes the way the Webber system restructures and reformats the template mapping system. The plain language of this section reveals that Webber simply does not use or even mention using any "pattern descriptors," (by that name or any other), in this process. By asserting the opposite, the examiner is in effect asking the applicant to prove the negative.

First, the examiner asserts that Webber provides a teaching. Then, the examiner provides a location where the asserted teaching is supposedly set forth. However, when the actual language is read, it does not provide the teaching asserted by the examiner. The examiner has simply misread the Webber reference, as Webber does not provide a teaching of "receiving one or more templates, each of said templates having said at least one pattern descriptor." The applicant can only point out that the reference does not say what the examiner asserts.

Instead, Webber simply accesses the table field layouts from the sending computer, which are then dynamically input into the template mapping system. While this limits the types of data structures that can be input into the Webber system, no intervention from the graphical user interface is required. Because the first computer in Webber is already configured to recognize the data structures given as input, the Webber system automatically configures the template mapping system with no input from a user. Webber makes this point explicitly at col. 7, lines 13-21:

> The table field layouts 40 are typically supplied by the sending computer and are dynamically input to the template mapping system 10 for use in processing the input dataset. The table field layouts 40 describe the names and maximum sizes of the various fields (in bytes) for each table. FIG. 3 is a diagram showing the general components of a table field layout (at left) mapped to an actual exemplary table field layout (at right). The table field layout is for a manifest history and manifest summary dataset.

In contrast, the present invention is configured to be able to handle patterns in the data structures that are NOT already known (or "manifest" to use the language of Webber). The drawback of the present invention, when compared to Webber, is that the unlike Webber, the present invention does not generate the template mapping system with no input from the user. The advantage of the present invention, when compared to Webber, is that the present invention is able to translate a much broader range of data into a common format, because the pattern descriptor is received from a graphical user interface. It is for this reason that claims 1, 9 and 15 of the present invention (and all remaining claims by virtue of dependency) includes the step of "receiving at least one pattern descriptor selected from a graphical user interface." This step is simply not present in Webber.

The examiner again mischaracterizes the Webber reference where the examiner recites that Webber discloses the step of "receiving at least one pattern descriptor selected from a graphical user interface" and references the language "field descriptions 400" at col. 7, lines 9-38. This portion of the Webber reference again does not support the examiner's contention. The "field descriptions" described by Webber are not input using a graphical user interface. As stated by Webber,

"The table field layout is for a manifest history and manifest summary dataset" col. 7, line 20.

and

> "The table definition 150 comprises three components: 1) comments and notes 200 (optional); 2) a table description 300; and 3) field descriptions 400 which provide details of fields within that table. The table definition 150 can be repeated for as many tables as required… Col. 7, line 22-25.
>
> The field descriptions 400 include the field identifier (column 1), the field names (column 2), size (column 3), data format (column 4), and justification parameters (column 5) which may be needed by the mapping process to determine the physical format of the table." Col. 7, line 33-38.

Accordingly, as described by Webber, it is the "field descriptions" that provide details of the fields within the table. Contrary to the examiner's assertions, these descriptions are NOT provided by a selection from a graphical user interface. The teaching of "receiving at least one pattern descriptor selected from a graphical user interface" is simply not present in the Webber disclosure, and the specific portions of the record that the examiner references to support the examiner's assertions to the contrary do not provide the disclosure that the examiner describes.

With respect to Lennon, Lennon is even further afield from the present invention. The Lennon reference describes a method for browsing electronically-accessible resources using descriptions of the resources, wherein the descriptions of the resources have descriptor components. As such, the Lennon reference has absolutely nothing whatsoever to do with extracting and converting data from one or more information sources into a common format. The only way that the examiner can even assert that Lennon shows any aspect of the applicant's claims is by mischaracterizing Lennon's teaching on the process by which Lennon "selects a pattern descriptor." Toward that end, the examiner alleges that Lennon provides the teaching of "receiving at least one pattern descriptor selected from a graphical user interface." The examiner specifically recites that Lennon teaches "pattern descriptors are selected as a tool to perform processing such as transformations, presentations, etc." and references column 15, lines 65 through 16, line 17 as evidence of this teaching. However, the referenced section of Lennon reads as follows:

> The preferred DDF also uses an API for the processing of descriptions. This enables applications and tools to perform further processing (eg., transformations, presentations, etc.) on serialised descriptions. The preferred API, which is described further in Section 2.3, is based on the Document Object Model called the DOM, which has been standardised by the W3C for use with XML documents.
>
> The DesOM API also enables the application of rule-based processing, which can be used to: Extend a description by inferring the presence of additional descriptors based on the existence or absence of stored descriptors; Influence/control the presentation of a description; Select descriptions or components of descriptions; Translate a stored description into another language on the basis of requirement; Transform a description to use a new description scheme. This rule-based processing is described in more detail in Sections 7 to 11.

This section of Lennon thus shows an *automated* processing of descriptions by the DDF using an API. Lennon's processing has nothing whatsoever to do with input from a GUI. Instead, Lennon is describing "rule-based processing" that is used to manipulate the descriptions in an automated fashion. The examiner has thus again mischaracterized the teaching of the reference. Just as Webber did not teach that "the pattern descriptor is selected," neither does Lennon.

With respect to the limitation of claims 8, 14, and 22, the examiner concedes that Webber does not teach "storage bins consisting of an input bin, a wait bin, an incomplete bin and a complete bin." The examiner then asserts that Lennon provides this teaching because Lennon teaches that "descriptors can be complex data types that can be represented in a hierarchical fashion such as in bins." This statement mischaracterizes the teaching of Lennon. The examiner then references col. 15, lines 51-64. This section of Lennon reads as follows:

> The object model provides the core semantics of the description and is based on the descriptor entity. This model has the advantage that the containment relationship is inherent in the model. This containment relationship is particularly important in the description of audiovisual resources for two reasons. First, the structure of many audiovisual resources has an inherent hierarchical structure (eg., a video clip contains shots which contain key frames, etc.). Second, the representation values for many descriptors can be complex datatypes that can be represented in a hierarchical fashion (eg., a histogram contains bins which contain frequencies). The object model of the preferred DDF is called the Description Object Model DesOM). It is discussed in Section 2.2.

Thus, while Lennon does use the word "bins" it is clear that Lennon is using the word in the context of a histogram. A histogram is a graphical representation, showing a visual impression of the distribution of experimental data. A histogram consists of tabular frequencies, shown as adjacent rectangles, erected over discrete intervals which are referred to as "bins," with each "bin" having an area equal to the frequency of the observations in the interval. Thus, in the context of a histogram, (which is the explicit context used by Lennon) a "bin" is simply a discrete interval. That is what Lennon is saying.

In contrast, the present application uses the term "bins" to describe sections of the data that have been stored in separate locations. As set forth in the specification at paragraph 0051:

> [0051] At block 264, the structured data having a common format is generated using the method described above. The generated structured data can then be stored in the wait bin, the incomplete bin, or a complete bin. The waiting bin permits the user to view files that matched required items in a template, thereby permitting the user to manually revise the pattern descriptor for a modified template or to designate the file as complete. The incomplete bin lists all files where no direct matches were found with the available templates. For files in the incomplete bin, the user views these files and creates templates to parse these "incomplete" files, and uses new templates to reprocess any failed files. The complete bin lists files that have been success-

fully parsed and the template that was used to parse it. Additionally, for each storage bin the user has the ability to generate statistical information.

Thus, contrary to the examiner's suggestion, the use of the word "bins" as described in Lennon is not the same as the use of the word "bins" as described in the present application.

Ascertaining The Differences Between The Claimed Invention And The Prior Art

The examiner has rejected claims 1-22, and 29-31 under 35 USC 103(a) as being obvious over Webber, US Patent No. 5,909,570 in view of Lennon, US Patent No. 7,287,018. As shown above, in setting forth the reasons for rejecting these claims, the examiner mischaracterizes the teaching of both the Webber and Lennon references.

The present invention differs from the Webber and Lennon references because the template mapping system of Webber is not configured to use a pattern descriptor. As shown above, the Webber method is configured in advance to recognize the patterns in the data structures of the data used as input. This is possible because, as Webber explicitly describes, it is treated as a given that Webber's system "knows" the format of the data that is being input into Webber's system, and Webber's system is configured in advance to receive data in that pre-determined format. Webber does this by accessing the table field layouts from the sending computer, which are then dynamically input into the template mapping system. As such, contrary to the examiner's suggestion, Webber does not include the disclosure of one or more templates, each having at least one pattern descriptor. The Lennon reference also does not include use one or more templates, each having at least one pattern descriptor, and the examiner has not asserted that Lennon contains this teaching. Accordingly, this limitation is not present in the prior art of record.

The present invention also differs from the Webber and Lennon references because neither the Webber or Lennon references disclose the step of "receiving at least one pattern descriptor selected from a graphical user interface." The limitation of "receiving at least one pattern descriptor selected from a graphical user interface," is required by claims 1, 9, 15, 22, and all remaining claims by virtue of dependency. As shown above, this step is not present in the Webber disclosure. The examiner's assertions to the contrary are simply not supported by the Webber reference. As shown above, the specific portion of the Webber reference cited by the examiner simply do not provide the disclosure that the examiner describes.

The Lennon reference also does not provide the teaching of "receiving at least one pattern descriptor selected from a graphical user interface." As shown above, Lennon only shows an *automated* processing of descriptions by the DDF using an API. Lennon's processing does not receive input from a GUI. Instead, Lennons "rule-based processing" manipulates the descriptions in an automated fashion. Just as Webber did not teach that "receiving at least one pattern descriptor selected from a graphical user interface," neither does Lennon.

With respect to the limitation of "storage bins consisting of an input bin, a wait bin, an incomplete bin and a complete bin" as set forth in claims 8, 14, and 22, the examiner concedes that Webber does not teach "storage bins consisting of an input bin, a wait bin, an incomplete bin and a complete bin." Contrary to the examiner's suggestion, Lennon also does not provide this teaching. As shown above, the examiner's suggestion that Lennon teaches that "descriptors can be complex data types that can be represented in a hierarchical fashion such as in bins" is not an accurate description of the Lennon disclosure. While Lennon discloses that the representation values for many descriptors can be complex datatypes that can be represented in a hierarchical fashion, Lennon only describes bins as a "data type," (eg., a histogram contains bins which contain frequencies), and not as a place in memory where discrete data is stored.

Thus, while Lennon does use the word "bins" it is clear that Lennon is using the word in a manner completely distinct from the limitations contained within claims 8, 14 and 22 of the present invention. The present application uses the term "bins" to describe locations where generated structured data may be stored. As set forth in the specification at paragraph 0051, generated structured data can then be stored in the wait bin, the incomplete bin, or a complete bin. Even if Lennon's "bins" were somehow stretched to include locations where generated structured data may be stored (as opposed to a data type, such as a discrete intervals in a histogram), there is no disclosure whatsoever to support the contention that these "bins" would include an input bin, a wait bin, an incomplete bin and a complete bin, as is required by the limitations of claims 8, 14 and 22 of the present invention. Indeed, the examiner has not even asserted as such.

Resolving The Level Of Ordinary Skill In The Pertinent Art

The USPTO guidelines are intended to assist Office personnel to make a proper determination of obviousness under 35 U.S.C. 103 and to provide an appropriate supporting rationale in view of the recent decision by the Supreme Court in *KSR International Co. v. Teleflex Inc.* (KSR), 550 U.S. 398, 82 USPQ2d 1385 (2007). The guidelines are based on the Office's current understanding of the law, and are believed to be fully consistent with the binding precedent of the Supreme Court. MPEP 2141 The Supreme Court reaffirmed principles based on its precedent that "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." *Id.* In determining the differences between the prior art and the claims, the question under 35 U.S.C. 103 is not whether the differences themselves would have been obvious, but whether the claimed invention as a whole would have been obvious. *Stratoflex, Inc. v. Aeroquip Corp.*, 713 F.2d 1530, 218 USPQ 871 (Fed. Cir. 1983); *Schenck v. Nortron Corp.*, 713 F.2d 782, 218 USPQ 698 (Fed. Cir. 1983)

In this case, the invention as a whole cannot be considered obvious because the differences between the claimed invention and the prior art of record consist of features that are missing entirely from the prior art of record. Accordingly, the examiner cannot even demonstrate that the elements of the instant claims are "familiar" or that the limitations are "known methods." As such, when considered as a whole, all claims of the present invention contains features that are missing entirely from the prior art of record.

Claims 1-22 and 29-31 all require one or more templates, each having at least one pattern descriptor. The Webber reference does not include the use one or more templates, each having at least one pattern descriptor. For the reasons shown above, the examiner's contention to the contrary is incorrect. The examiner has not asserted that Lennon contains this teaching. Thus, the prior art of record does not teach this limitation, and no prima facie case of obviousness can be established.

Claims 1-22 and 29-31 all require "receiving at least one pattern descriptor selected from a graphical user interface." The Webber reference does not include the step of "receiving at least one pattern descriptor selected from a graphical user interface." For the reasons shown above, the examiner's contention to the contrary is incorrect. The Lennon reference also does not include the step of "receiving at least one pattern descriptor selected from a graphical user interface." For the reasons shown above, the examiner's contention to the contrary is again incorrect. Thus, the prior art of record does not teach this limitation, and no prima facie case of obviousness can be established.

Claims 8, 14, and 22, all require the limitation of "storage bins consisting of an input bin, a wait bin, an incomplete bin and a complete bin." The examiner concedes that the Webber reference does not include the limitation of "storage bins consisting of an input bin, a wait bin, an incomplete bin and a complete bin." Contrary to the examiner's suggestion, the Lennon also does not provide this teaching. Lennon only describes bins as a "data type," (eg., a histogram contains bins which contain frequencies), and not as a place in memory where discrete data is stored. While Lennon does use the word "bins," Lennon is using the word in a manner completely distinct from the present invention. Thus, the prior art of record does not teach this limitation, and no prima facie case of obviousness can be established.

Conclusion

The prior art of record does not teach, either alone or in combination, the limitations of all rejected claims. Accordingly, the rejection of claims 1-22 and 29-31 under 35 U.S.C. § 103(a) should be withdrawn and all claims should be allowed.

Respectfully submitted,

Law Office of Douglas. E. McKinley, Jr.
PO Box 202
Richland WA 99352
Telephone: (509) 628-0809
Facsimile: (509) 628-2307

By / Douglas E. McKinley, Jr. /
Douglas E. McKinley, Jr.
Registration No. 40,280

Appendix

Listing of Claims Subject to Appeal

Claim Or Claims

The invention claimed is:

What is claimed is:

1. (Rejected) A method for extracting and converting data in a computer system from one or more information sources into a common format, comprising:

receiving said information sources in a computer system;

receiving at least one pattern descriptor selected from a graphical user interface;

receiving one or more templates, each of said templates having said at least one pattern descriptor;

applying said one or more templates to said information sources;

generating said data in a common format by parsing said information sources with a universal parsing agent that utilizes said one or more templates; and

storing said data in said common format.

2. (Rejected) The method of claim 1 wherein after storing said data in said common format, said method further comprises communicating said data to an application configured to process said common format.

3. (Rejected) The method of claim 2 wherein said application is a database application.

4. (Rejected) The method of claim 1 wherein said common format for said structured data is a Extensible Markup Language (XML) format.

5. (Rejected) The method of claim 1 wherein before receiving said one or more templates, said method further comprises,

generating said one or more templates by selecting a file from said information sources, and having a user select one or more pattern descriptors to describe said file.

6. (Rejected) The method of claim 5 further comprising permitting said user to define said one or more pattern descriptors.

7. (Rejected) The method of claim 1 wherein before receiving said one or more templates, said method further comprises permitting said user to select one or more templates from a template library.

8. (Rejected) The method of claim 1 wherein said storing of said data in said common format is selected from a group of storage bins consisting of an input bin, a wait bin, an incomplete bin, and a complete bin.

9. (Rejected) A system for extracting and converting data from one or more information sources into a common format, comprising:

a memory configured to receive said information sources, said memory configured to store one or more templates wherein each of said templates has at least one pattern descriptor selected from a graphical user interface;

an input device configured to receive said at least one pattern descriptor from a user interacting with a graphical user interface;

a processor programmed to:

apply said one or more templates to said information sources;

generate said data in said common format by parsing said information sources

with a universal parsing agent that utilizes said one or more templates; and

communicate said data in said common format.

10. (Rejected) The system of claim 9 wherein said processor is configured to communicate said data in said common format to an application configured to process said common format.

11. (Rejected) The system of claim 10 wherein said application is a database application.

12. (Rejected) The system of claim 9 wherein said common format for said structured data is a Extensible Markup Language (XML) format.

13. (Rejected) The system of claim 9 wherein said memory stores a template library from which a user can select one or more templates.

14. (Rejected) The system of claim 9 wherein said memory stores said data in said common format in a storage bin selected from a group of storage bins consisting of an input bin, a wait bin, and incomplete bin, and a complete bin.

15. (Rejected) A computer readable medium encoded with a computer program having computer-executable instructions for performing a method for extracting and converting data from one or more information sources into a common format, comprising:

receiving said information sources;

receiving at least one pattern descriptor selected from a graphical user interface;

receiving one or more templates, each of said templates having said at least one pattern descriptor;

applying said one or more templates to said information sources;

converting said data into said common format by parsing said information sources with a universal parsing agent that utilizes said one or more templates; and

storing said data in said common format.

16. (Rejected) The computer readable medium encoded with a computer program having computer-executable instructions for performing said method of claim 15 wherein after storing said data in said common format, said method further comprises communicating said data in said common format to an application configured to process said common format.

17. (Rejected) The computer readable medium encoded with a computer program having computer-executable instructions for performing said method of claim 16 wherein said application is a database application.

18. (Rejected) The computer readable medium encoded with a computer program having computer-executable instructions for performing said method of claim 15 wherein said common format for said structured data is a Extensible Markup Language (XML) format.

19. (Rejected) The computer readable medium encoded with a computer program having computer-executable instructions for performing said method of claim 15 wherein before receiving said one or more templates, said method further comprises,

generating said one or more templates by selecting a file from said plurality of information sources, and

having a user select one or more pattern descriptors to describe said file.

20. (Rejected) The computer readable medium encoded with a computer program having computer-executable instructions for performing said method of claim 16 further comprising permitting said user to define said one or more pattern descriptors.

21. (Rejected) The computer readable medium encoded with a computer program having computer-executable instructions for performing said method of claim 15 wherein before receiving said one or more templates, said method further comprises permitting said user to select one or more templates from a template library.

22. (Rejected) The computer readable medium encoded with a computer program having computer-executable instructions for performing said method of claim 15 wherein said storing of said data in said common format is selected from a group of storage bins consisting of an input bin, a wait bin, an incomplete bin, and a complete bin.

23. - 28. (Withdrawn)

29. (Previously Added) The method of claim 1 wherein said information sources are selected from the group of structured information sources, semi-structured information sources, unstructured information sources and combinations thereof.

30. (Previously Added) The system of claim 9 wherein said information sources are selected from the group of structured information sources, semi-structured information sources, unstructured information sources and combinations thereof.

31. (Rejected) The computer readable medium encoded with a computer program having computer-executable instructions for performing said method of claim 15 wherein information sources are selected from the group of structured information sources, semi-structured information sources, unstructured information sources and combinations thereof.

32. (Withdrawn)

Evidence Appendix

None.

Related Proceedings Appendix

There are no related proceedings.